An Algebraic Method for Compressing Very Large Symbolic Data Tables

Tzitzikas, Yannis

Published in:
Proceedings of the Workshop on Symbolic and Spatial Data Analysis (SSDA) of ECML/PKDD 2004

Publication date:
2004

Workshop on Symbolic and Spatial Data Analysis (SSDA'2004) of ECML/PKDD 2004, Pisa, Italy, September 2004

An Algebraic Method for Compressing Very Large Symbolic Data Tables

Yannis Tzitzikas

Institut d'Informatique
F.U.N.D.P. (University of Namur)
Rue Grandgagnage 21, B-5000, Belgium
Email : ytz@info.fundp.ac.be

Abstract. Although symbolic data tables summarize huge sets of data they can still become very large in size. This paper proposes a method for compressing a symbolic data table using the recently emerged *Compound Term Composition Algebra*. One charisma of CTCA is that the closed world hypotheses of its operations can lead to a remarkably high "compression ratio". The compacted form apart from having much lower storage space requirements, it allows designing more efficient algorithms for symbolic data analysis.

1 Introduction

As recent surveys state[1], the world produces between 1 and 2 exabytes (2^{60} bytes) of unique information per year, 90% of which is digital and with a 50% annual growth rate. Undoubtedly, this is a boon rather than a anathema. In addition, this plethoric growth rate has stimulated the development of new techniques and automated tools for assisting the transformation of large amounts of data into useful information and knowledge (see data mining and knowledge discovery in databases). *Symbolic data analysis* [3, 4] has been introduced in order to solve the problem of the analysis of data that are given on an aggregated form, i.e. where quantitative variables are given by intervals and where categorical variables are given by histograms. This kind of data are generated when we summarize huge sets of data. Inescapably, even a symbolic data table could become very large in size, making its management problematic in terms of both storage space and computational time.

This paper aims to convey some recent advances from the area of knowledge representation (in particular from the area of faceted taxonomies and faceted classification), that could be exploited for symbolic data analysis. Specifically, this paper gives the theoretical foundation of a novel method that can be used to *compress* (i.e. to reduce the storage space requirements) of large symbolic data

[1] http://www.sims.berkeley.edu/research/projects/how-much-info-2003/

tables. The proposed compression is lossless i.e. from the compressed form we can infer exactly what we can from the original symbolic data table.

The contribution of the method is not exhausted to storage space minimization as the resulting compact form could allow the design more efficient symbolic analysis algorithms.

For reasons of space, this paper describes only the principles of this method and gives some indicative examples. The interested reader is referred to the references that are given. The rest of this paper is organized as follows. Section 2 sketches the idea and Section 3 recalls the basics of the *Compound Term Composition Algebra* (CTCA), upon which the proposed method is founded. Subsequently, Section 4 describes in more detail the steps of this technique and Section 5 gives some indicative examples of compression using CTCA. Finally, Section 6 concludes the paper and identifies issues for further research.

2 The Idea

A *Symbolic data table* is a table of data where the columns are the *symbolic variables* which are used in order to describe a set of units called *individuals*. Rows are called *symbolic descriptions* of these individuals because they are not as usual, only tuples of single quantitative or categorical values. For instance, the values of the cells can be intervals (if the variable is quantitative) or frequency distributions (if the variable is categorical). Recall that in classical data analysis a cell can have a single quantitative or categorical value. In general, we could distinguish variables according to their range to (a) single quantitative (e.g. age=18), (b) single categorical (or taxonomic) (e.g. color=red), (c) multi-valued quantitative or categorical (e.g. age=$\{11, 18\}$, color=$\{red, green\}$) (d) interval (e.g. age=[10,20]), and (e) multi-valued with weights (e.g. histograms). Clearly, (a) and (b) are special cases of (c), while (c) is special case of (e) (i.e. when all weights are either 0 or 1) for more see [4].

This paper proposes a method for compacting a symbolic data table by exploiting the *Compound Term Composition Algebra* ($CTCA$). CTCA is a recently emerged algebra that allows specifying the *valid* (meaningful) *compound terms* (conjunction of terms) over a *faceted taxonomy* in a flexible and efficient manner (for more see [13, 12]). A system around CTCA has already been developed (FASTAXON [14]) and there has already been proposed a Web annotation language that allows exchanging faceted taxonomies and expressions of CTCA (for more see XFML+CAMEL [2]). In brief, a faceted taxonomy is a set of taxonomies each one describing the domain of interest from a different (preferably orthogonal) point of view (for more about faceted classification and analysis see [10, 5, 15, 6, 7]). Faceted taxonomies are used in Web Catalogs, Libraries [7], Software Repositories [8, 9], and several others application domains. Current interest in faceted taxonomies is also indicated by several recent or ongoing projects (like FATKS[2],

[2] http://www.ucl.ac.uk/fatks/database.htm

FACET[3], FLAMENGO[4]) and the emergence of XFML [1](Core-eXchangeable Faceted Metadata Language) a markup language for applying the faceted classification paradigm on the Web. Having a faceted taxonomy each domain object (e.g. book or Web page) can be indexed using a *compound term*, i.e., a set of terms containing one or more terms from each facet. We shall use the term *materialized faceted taxonomy* to refer to a faceted taxonomy accompanied by a set of object indices. For example, Figure 1 shows a very small but indicative faceted taxonomy consisting of three facets that is appropriate for indexing hotel Web pages.

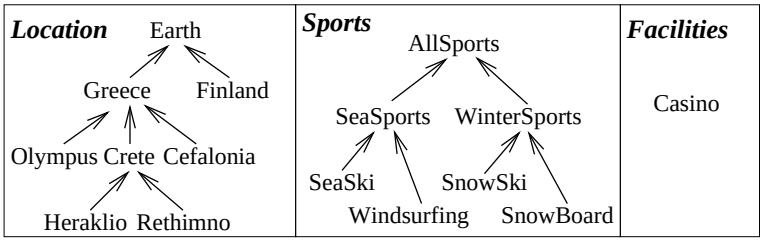

Fig. 1. A faceted taxonomy for indexing hotel Web pages

Roughly, and according to the above perspective and phraseology, each symbolic data table can be viewed as a *materialized faceted taxonomy*. This analogy is not hard to grasp. Each symbolic variable can be viewed as a *facet*. Now the range of each symbolic variable can be viewed as a *taxonomy*, i.e. as a partially ordered set of terms (clearly, categories, intervals, and subsets are partially ordered domains). Now each row of the symbolic data table can be viewed as an *object* that has been *indexed* according to a faceted taxonomy, i.e. as an object that has been associated with a *compound term* of the faceted taxonomy, i.e. with a set of values from the range of the symbolic variables.

Several algorithms for finding an expression of CTCA that describes those compound terms that are *extensionally valid* in a materialized faceted taxonomy were given and analyzed in [11]. In other words, these algorithms mine an expression of CTCA that specifies the set of all distinct compound terms that are meaningful, where a compound term is considered meaningful if it is applicable to at least one object of the object base. It follows, that the same algorithms can be exploited for the problem at hand, specifically for finding a short (in storage space) expression of CTCA that specifies the rows of a symbolic data table.

Specifically, this paper focuses on symbolic variables with partially ordered ranges, i.e. taxonomically-ordered categorical, multi-valued quantitative or categorical, and interval-valued variables. The reason is that in this case the employment of CTCA yields remarkably high compression ratios. However, CTCA can be applied even on unordered ranges, i.e. on sets (we can view a set as a

[3] http://www.glam.ac.uk/soc/research/hypermedia/facet_proj/index.php
[4] http://bailando.sims.berkeley.edu/flamenco.html

poset with an empty ordering relation), so the proposed method can be also applied on variables whose range is a set of histograms. However, an issue for further research is to investigate ordering relations over histograms because their availability would allow obtaining higher compression ratios even for this kind of variables (especially when lossy compression is tolerable).

3 Faceted Taxonomies and the *Compound Term Composition Algebra*

Table 1 below recalls in brief the basic notions around taxonomies, faceted taxonomies and materialized faceted taxonomies (for more please refer to [13]).

Name	Notation	Definition
terminology	$\mathcal{T}$	a set of names called terms
subsumption	$\leq$	a preorder relation (reflexive and transitive)
taxonomy	$(\mathcal{T}, \leq)$	$\mathcal{T}$ is a terminology, $\leq$ a subsumption relation over $\mathcal{T}$
faceted taxonomy	$\mathcal{F}=\{F_1, ..., F_k\}$	$F_i = (\mathcal{T}_i, \leq_i)$, for $i = 1, ..., k$ and all $\mathcal{T}_i$ are disjoint
compound term over $\mathcal{T}$	s	any subset of $\mathcal{T}$ (i.e. any element of $\mathcal{P}(\mathcal{T})$)
compound terminology	S	a subset of $\mathcal{P}(\mathcal{T})$ that includes $\emptyset$
compound ordering	$\preceq$	$s \preceq s'$ iff $\forall t' \in s' \ \exists t \in s$ such that $t \leq t'$.
broaders of s	$\mathrm{Br}(s)$	$\{s' \in P(\mathcal{T}) \mid s \preceq s'\}$
narrowers of s	$\mathrm{Nr}(s)$	$\{s' \in P(\mathcal{T}) \mid s' \preceq s\}$
broaders of S	$Br(S)$	$\cup\{\mathrm{Br}(s) \mid s \in S\}$
narrowers of S	$Nr(S)$	$\cup\{\mathrm{Nr}(s) \mid s \in S\}$
object domain	Obj	any denumerable set of objects
interpretation of $\mathcal{T}$	I	any function $I : \mathcal{T} \rightarrow 2^{Obj}$
model of $(\mathcal{T}, \leq)$ induced by I	$\bar{I}$	$\bar{I}(t) = \cup\{I(t') \mid t' \leq t\}$
materialized faceted taxonomy	$(\mathcal{F}, I)$	$\mathcal{F}$ is a faceted taxonomy $\{F_1, ..., F_k\}$, I is an interpretation of $\mathcal{T} = \bigcup_{i=1,k} \mathcal{T}_i$

Table 1. Notations

CTCA was proposed for defining the meaningful compound terms over a faceted taxonomy in a flexible and efficient manner. The problem of meaningless compound terms and the effort needed to specify the meaningful ones is a practical problem identified even by Ranganatham himself [10] (80 years ago) and it is probably the main reason why faceted taxonomies have not dominated every application domain despite their uncontested advantages over the single-hierarchical taxonomies. CTCA is the only well-founded and flexible solution to this problem.

CTCA has four basic algebraic operations, namely, *plus-product* ($\oplus$), *minus-product* ($\ominus$), *plus-self-product*, ($\overset{*}{\oplus}$), and *minus-self product* ($\overset{*}{\ominus}$). They are all

operations over $\mathcal{P}(\mathcal{T})$, the powerset of $\mathcal{T}$, where $\mathcal{T}$ is the union of the terminologies of all facets. The initial operands, thus the building blocks of the algebra, are the basic compound terminologies, which are the facet terminologies with the only difference that each term (for reasons of notational simplicity) is viewed a singleton. Specifically, the *basic compound terminology* of a terminology $\mathcal{T}_i$ is defined as: $T_i = \{\{t\} \mid t \in \mathcal{T}_i\} \cup \{\emptyset\}$. If e is an expression, S_e denotes the outcome of this expression and is called the *compound terminology* of e. An expression e over $\mathcal{F}$ is defined according to the following grammar ($i = 1, ..., k$):

$$e ::= \oplus_P(e, ..., e) \mid \ominus_N (e, ..., e) \mid \overset{*}{\oplus}_P T_i \mid \overset{*}{\ominus}_N T_i \mid T_i,$$

where the parameters P and N denote sets of valid and invalid compound terms over the range of the operation, respectively. Roughly, CTCA allows specifying the valid compound terms over a faceted taxonomy by providing a small set of valid (P) and a small set of invalid (N) compound terms. The self-product operations allow specifying the meaningful compound terms over one facet. Specifically, the definition of each operation of CTCA is summarized in Table 2 where S, S' denote compound terminologies. In addition, $(S_e, \preceq)$ is called the *compound taxonomy* of e. The associated inference mechanism and the closed world assumption of each operation, makes the task of specifying the meaningful compound terms flexible and fast. The algorithm given in [13] takes as input a expression e and a compound term s, and checks whether $s \in S_e$. This algorithm has polynomial time complexity, specifically $O(|\mathcal{T}|^3 * |\mathcal{P} \cup \mathcal{N}|)$, where $\mathcal{P}$ denotes the union of all P parameters and $\mathcal{N}$ denotes the union of all N parameters appearing in e.

Operation	e	S_e
product	$S_1 \oplus ... \oplus S_n$	$\{ s_1 \cup ... \cup s_n \mid s_i \in S_i\}$
plus-product	$\oplus_P(S_1, ...S_n)$	$S_1 \cup ... \cup S_n \ \cup \ Br(P)$
minus-product	$\ominus_N(S_1, ...S_n)$	$S_1 \oplus ... \oplus S_n - Nr(N)$
self-product	$\overset{*}{\oplus} (T_i)$	$P(\mathcal{T}_i)$
self-plus-product	$\overset{*}{\oplus}_P (T_i)$	$T_i \cup Br(P)$
self-minus-product	$\overset{*}{\ominus}_N (T_i)$	$\overset{*}{\oplus} (T_i) - Nr(N)$

Table 2. The operations of the Compound Term Composition Algebra

For example, Table 3 (that is found on the appendix) shows the partition of the compound terms of the faceted taxonomy of Figure 1 into the set of valid and the set of invalid compound terms. Instead of defining this partition explicitly, with CTCA one can define it in a more flexible and quick manner. Specifically, this partition can be specified by the subsequent expression:

$$e = (Location \ominus_N Sports) \oplus_P Facilities$$

with the following P and N parameters:

$$N = \{\{Crete, WinterSports\}\},$$

$$P = \begin{array}{l} \{Cefalonia, WinterSports\}\} \\ \{\{Cefalonia, SeaSki, Casino\}, \\ \{Cefalonia, Windsurfing, Casino\}\} \end{array}$$

CTCA can be exploited both forthrightly *and* reversely, i.e. a designer can formulate an expression in order to specify quickly the desired set of compound terms, while from an existing set of compound terms an algorithm can find an expression that describes these compound terms. It is the latter direction that is appropriate for symbolic data analysis.

In order to apply CTCA for compacting symbolic data tables we only have to consider facets that range a set of *intervals*. This is rather a trivial extension, as we can consider each interval [a,b] as a term. The ordering between interval terms can be inferred easily, i.e. $[a, b] \leq [c, d]$ iff $c \leq a$ and $b \leq d$, and there is no need for storing these relationships. So CTCA applies on intervals as it is.

Note that the disjointness of facet terminologies can be implemented in practise by prefixing each value of the range of a variable by the variable name.

4 The Technique

Roughly, a symbolic data table with k columns and n rows can be compressed in three steps:

(a) At first, we organize the range of each variable as a partially ordered set (poset) and we store it.
Note that if the range of a variable is a set of categories that are partially ordered, i.e. a taxonomy, then it is enough to store only the transitive reduction of the taxonomic ordering. If the range of a variable is a set R of subsets of a set D (i.e. $R \subseteq \mathcal{P}(D)$ where $\mathcal{P}(D)$ denotes the powerset of D), then we again have a poset, i.e. the partially ordered set $(R, \subseteq)$. In this case we only have to store R as here the ordering relation corresponds to the relation $\subseteq$ which can be deduced algorithmically (for any two sets s and s' we can check whether $s \subseteq s'$). Finally, if the range of a variable is a set of intervals L then we only have to store L because again the ordering relation can be deduced.

(b) Subsequently, we can run one of the algorithms described in [11] that *mine* an expression of CTCA that describes exactly the rows of our table.
Using the notations of the previous section, the objective of these algorithms is to find an expression e such that

$$S_e = \{s \in \mathcal{P}(\mathcal{T}) \mid \bar{I}(s) \neq \emptyset\}$$

where if $s = \{t_1, ..., t_k\}$ then $I(s) = I(t_1) \cap ... \cap I(t_k)$. That paper gives the algorithms for two straightforward methods for extracting a plus-product and a minus-product expression and an exhaustive algorithm for finding the *shortest* (i.e. the most space economical) expression. The latter yields expressions with remarkably low storage space requirements, thanks to the

closed-world assumptions of CTCA, but its computational complexity if remarkably higher. For reasons of space the description of the algorithms is omitted.

(c) Finally, we store the mined expression and its parameters (e.g. in a relational database as it has been done in FASTAXON [14]).

After the above process we can delete the symbolic data table and keep stored only the posets and the mined expression e. Now suppose that we want to check whether an arbitrary tuple s (over the domain of our variables) exists in the table. We don't have to restore the initial table in order to answer this question. Instead, we run the algorithm described in [13] which takes as input a faceted taxonomy, an expression e and a compound term s and decides in polynomial time whether $s \in S_e$.

Another remark that should be mentioned here is that it is also possible to *browse* the symbolic table without having to reconstruct it. Specifically, by the faceted taxonomy $\mathcal{F}$ and the expression e we can derive dynamically a navigation tree that allows browsing all compound terms in S_e using the algorithm described in [13] that has been implemented in FASTAXON [14].

Of course, at any time we could run a (quite simple) algorithm for reconstructing the symbolic data table at its original form.

5 Indicative Examples

This section presents a small number of intuitive examples for demonstrating the potential of CTCA for the problem at hand.

Consider that we have two variables A and B. The variable A ranges over the set $\{a_1, a_2, a_3\}$ and assume that this set is ordered according to a taxonomic relation (subsumption) as follows: $a_3 \leq a_2 \leq a_1$. Now consider the following table

A	B
a_1	b_1
a_2	b_1
a_3	b_1

The rows of this table can be described by the expression $e = A \oplus_P B$ where $P = \{\{a_3, b_1\}\}$. One can easily see that $S_e = \{\{a_1, b_1\}, \{a_2, b_1\}, \{a_3, b_1\}\}$.

Alternatively, they can be described by the expression $e' = A \ominus_N B$ where $N = \emptyset$ as $A \ominus_\emptyset B = A \oplus B = \{\{a_1, b_1\}, \{a_2, b_1\}, \{a_3, b_1\}\}$.

Now assume that the range of A is the taxonomy $(\{a_1, a_2, a_3, a_4\}, \{a_2 \leq a_1, a_3 \leq a_2, a_4 \leq a_2\})$, the range of B is the taxonomy $(\{b_1, b_2\}, \{b_2 \leq b_1\})$ and that we have the following table:

A	B
$\{a_3, a_4\}$	b_1
a_2	b_2
a_1	b_1
a_1	b_2
a_2	b_1
a_3	b_1

Here and in order to describe the set $\{a_3, a_4\}$ we are obliged to use a self-product operation over A. We can describe the rows of this table by any of the above three expressions:

- $e_1 = (\overset{*}{\oplus}_{P1}(A)) \oplus_{P2}(B)$ where $P1 = \{\{a_3, a_4\}\}$ and $P2 = \{\{a_3, a_4, b_1\}, \{a_2, b_2\}\}$
- $e_2 = (\overset{*}{\ominus}_{N1}(A)) \oplus_{P2}(B)$ where $N1 = \emptyset$ and $P2 = \{\{a_3, a_4, b_1\}, \{a_2, b_2\}\}$.
- $e_3 = (\overset{*}{\ominus}_{N1}(A)) \ominus_{N2}(B)$ where $N1 = \emptyset$ and $N2 = \{\{a_3, a_4, b_2\}\}$.

Clearly, e_3 is the most space economical expression as it requires us to keep stored only one compound term that consists of three single terms.

Example 1.

Assume that we have the following table with information about hotels:

Id	Location	Prices
H1	Heraklion	[30,50]
H2	Lixouri	[33,40]
H3	Heraklion	[25,300]
H4	Heraklion	[33,40]

For notational simplicity we shall use A for Location and B for Prices. The above table (by ignoring the first column) can be represented by the expression $e_1 = A \ominus_{N_1} B$ where $N_1 = \emptyset$ as all combinations between the domain of these two variables are valid (appear or are semantically inferred from those that appear). Specifically, although Lixouri does not co-appear in the table with neither [30,50] nor with [25, 300], these combination are valid because since there is a hotel at Lixouri with rates [33,40], it is true that we can find a hotel at Lixouri at [30,50] or [25,300] Euros.

Example 2.

Let us now modify one cell of the above table:

Id	Location	Prices
H1	Heraklion	[30,50]
H2	Lixouri	**[30,50]**
H3	Heraklion	[25,300]
H4	Heraklion	[33,40]

This table can be represented by the expression $e_2 = A \ominus_{N_2} B$ where $N_2 = \{\{Lixouri, [33, 40]\}\}$.

Example 3.

Let us now add one more row and one more column to the table of the previous example

Id	Location	Prices	SportsAndFacilities
H1	Heraklion	[30,50]	SeaSki, Sauna
H2	Lixouri	[30,50]	
H3	Heraklion	[25,300]	WindSurfing, SeaSki, Sauna
H4	Heraklion	[33,40]	
H5	Helsinki	[33,40]	SmokeSauna

Let C denote the variable SportsAndFacilities and let the range of C be organized as shown in Figure 2. Let's now try finding the expression that describes this table. The "subtable" that consists of the columns B and C is described by the expression e_2 as we saw earlier in Example 2. Now the range of variable C can be expressed using a self-product operation, specifically by $e_C = \overset{*}{\oplus}_{P_C} (C)$ where $P_C = \{\{WindSurfing, SeaSki, Sauna\}\}$. Note that if SmokeSauna did not belong to the range of C then we would have defined e_C as follows: $e_C = \overset{*}{\ominus}_{N_C} (C)$ with $N_C = \emptyset$.

In order to represent the whole table we have to combine e_2 and e_C. This can be obtained as: $e_3 = e_2 \oplus_{P_3} e_C$ where

$P_3 = \{\{Heraklion, [25, 300], Lixouri, \{WindSurfing, SeaSki, Sauna\}\}$

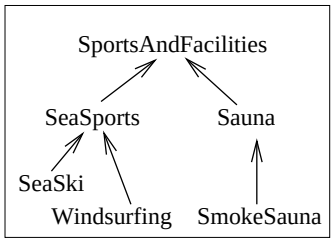

Fig. 2. The range of the symbolic variable SportsAndFacilities

Summarizing, CTCA can indeed compact a symbolic data table and can yield to remarkably high compression ratios. One can easily guess that the more symbolic variables we have and the more numerous are the ranges of these variables, the higher compression ratio we can achieve with CTCA.

6 Epilogue

Although symbolic data tables summarize huge sets of data they can still become very large in size. This paper proposes a method for compressing a symbolic

data table using the recently emerged Compound Term Composition Algebra (CTCA). One charisma of CTCA for the problem at hand is that the closed world hypotheses of its operations (described analytically at [12]) can lead to a remarkably high "compression ratio". Another remark that have to be mentioned here is that the functionality offered by CTCA cannot be obtained by using a classical logic-based formalism, like Description Logics, as it was shown in [12]. At last, but not least, this paper identified the analogies between symbolic data tables and faceted taxonomies (and CTCA) in order to act as a two-way canal between the two communities. An issue for further research is the characterization of the proposed approach according to Kolmogorov's complexity and the extension of this method for frequency-valued symbolic variables.

Acknowledgement

Many thanks to Tonia Dellaporta for the fruitful discussions on this issue and to Monique Noirhomme-Fraiture for providing me with very useful material for Symbolic Data Analysis.

References

1. "XFML: eXchangeable Faceted Metadata Language". http://www.xfml.org.
2. "XFML+CAMEL:Compound term composition Algebraically-Motivated Expression Language". http://www.csi.forth.gr/markup/xfml+camel.
3. H. H. Bock and E. Diday. *Analysis of Symbolic Data*. Springer-Verlag, 2000. ISBN: 3-540-66619-2.
4. Edwin Diday. "An Introduction to Symbolic Data Analysis and the Sodas Software". *Journal of Symbolic Data Analysis*, 0(0), 2002. ISSN 1723-5081.
5. Elizabeth B. Duncan. "A Faceted Approach to Hypertext". In Ray McAleese, editor, *HYPERTEXT: theory into practice, BSP*, pages 157–163, 1989.
6. P. H. Lindsay and D. A. Norman. *Human Information Processing*. Academic press, New York, 1977.
7. Amanda Maple. "Faceted Access: A Review of the Literature", 1995. http://theme.music.indiana.edu/tech_s/mla/facacc.rev.
8. Ruben Prieto-Diaz. "Classification of Reusable Modules". In *Software Reusability. Volume I*, chapter 4, pages 99–123. acm press, 1989.
9. Ruben Prieto-Diaz. "Implementing Faceted Classification for Software Reuse". *Communications of the ACM*, 34(5):88–97, 1991.
10. S. R. Ranganathan. "The Colon Classification". In Susan Artandi, editor, *Vol IV of the Rutgers Series on Systems for the Intellectual Organization of Information*. New Brunswick, NJ: Graduate School of Library Science, Rutgers University, 1965.
11. Yannis Tzitzikas and Anastasia Analyti. "Mining the Meaningful Compound Terms from Materialized Faceted Taxonomies ", 2004. (Submitted for publication in Knowledge and Information Systems Journal).
12. Yannis Tzitzikas, Anastasia Analyti, and Nicolas Spyratos. "The Semantics of the Compound Terms Composition Algebra". In *Procs. of the 2nd Intern. Conference on Ontologies, Databases and Applications of Semantics, ODBASE'2003*, pages 970–985, Catania, Sicily, Italy, November 2003.

13. Yannis Tzitzikas, Anastasia Analyti, Nicolas Spyratos, and Panos Constantopoulos. "An Algebraic Approach for Specifying Compound Terms in Faceted Taxonomies". In *Information Modelling and Knowledge Bases XV, 13th European-Japanese Conference on Information Modelling and Knowledge Bases, EJC'03*, pages 67–87. IOS Press, 2004.
14. Yannis Tzitzikas, Raimo Launonen, Mika Hakkarainen, Pekka Kohonen, Tero Leppanen, Esko Simpanen, Hannu Tornroos, Pekka Uusitalo, and Pentti Vanska. "FASTAXON: A system for FAST (and Faceted) TAXONomy design.". In *Proceedings of 23th Int. Conf. on Conceptual Modeling, ER'2004*, Shanghai, China, November 2004. (an on-line demo is available at http://fastaxon.erve.vtt.fi/).
15. B. C. Vickery. "Knowledge Representation: A Brief Review". *Journal of Documentation*, 42(3):145–159, 1986.

A Appendix

Valid		Invalid	
Earth, AllSports	Greece, AllSports	Crete, WinterSp.	Cefalonia, WinterSp.
Finland, AllSports	Olympus, AllSports	Rethimno, WinterSp.	Heraklio, WinterSp.
Crete, AllSports	Cefalonia, AllSports	Olympus, SeaSki	Olympus, WindSurf.
Rethimno, AllSports	Heraklio, AllSports	Crete, SnowBoard	Cefalonia, SnowBoard
Earth, SeaSports	Greece, SeaSports	Rethimno, SnowBoard	Heraklio, SnowBoard
Finland, SeaSports	Crete, SeaSports	Crete, SnowSki	Cefalonia, SnowSki
Cefalonia, SeaSports	Rethimno, SeaSports	Rethimno, SnowSki	Heraklio, SnowSki
Heraklio, SeaSports	Earth, WinterSp.	Finland, Cas.	Olympus, Cas.
Greece, WinterSp.	Finland, WinterSp.	Crete, Cas.	Heraklio, Cas.
Olympus, WinterSp.	Earth, SeaSki	Rethimno, Cas.	WinterSp., Cas.
Greece, SeaSki	Finland, SeaSki	SnowBoard, Cas.	SnowSki, Cas.
Crete, SeaSki	Cefalonia, SeaSki	Olympus, SeaSports	Crete, WinterSp., Cas.
Rethimno, SeaSki	Heraklio, SeaSki	Cefalonia, WinterSp., Cas.	Rethimno, WinterSp., Cas.
Earth, WindSurf.	Greece, WindSurf.	Heraklio, WinterSp., Cas.	Olympus, SeaSki, Cas.
Finland, WindSurf.	Crete, WindSurf.	Olympus, WindSurf., Cas.	Crete, SnowBoard, Cas.
Cefalonia, WindSurf.	Rethimno, WindSurf.	Cefalonia, SnowBoard, Cas.	Rethimno, SnowBoard, Cas.
Heraklio, WindSurf.	Earth, SnowBoard	Heraklio, SnowBoard, Cas.	Crete, SnowSki, Cas.
Greece, SnowBoard	Finland, SnowBoard	Cefalonia, SnowSki, Cas.	Rethimno, SnowSki, Cas.
Olympus, SnowBoard	Earth, SnowSki	Heraklio, SnowSki, Cas.	Olympus, AllSports, Cas.
Greece, SnowSki	Finland, SnowSki	Crete, AllSports, Cas.	Rethimno, AllSports, Cas.
Olympus, SnowSki	Earth, AllSports, Cas.	Heraklio, AllSports, Cas.	Crete, SeaSports, Cas.
Greece, AllSports, Cas.	Cefalonia, AllSports, Cas.	Rethimno, SeaSports, Cas.	Heraklio, SeaSports, Cas.
AllSports, Cas.	SeaSports, Cas.	Olympus, WinterSp., Cas.	Crete, SeaSki, Cas.
SeaSki, Cas.	Windsurf., Cas.	Rethimno, SeaSki, Cas.	Heraklio, SeaSki, Cas.
Earth, Cas.	Greece, Cas.	Crete, WindSurf., Cas.	Rethimno, WindSurf., Cas.
Cefalonia, Cas.	Earth, SeaSports, Cas.	Heraklio, WindSurf., Cas.	Olympus, SnowBoard, Cas.
Greece, SeaSports, Cas.	Earth, SeaSki, Cas.	Olympus, SnowSki, Cas.	Finland, AllSports, Cas.
Greece, SeaSki, Cas.	Cefalonia, SeaSki, Cas.	Finland, SeaSports, Cas.	Finland, WinterSp., Cas.
Earth, WindSurf., Cas.	Greece, WindSurf., Cas.	Finland, SeaSki, Cas.	Finland, WindSurf., Cas.
Cefalonia, WindSurf., Cas.	Cefalonia, SeaSports, Cas.	Finland, SnowSki, Cas.	Finland, SnowBoard, Cas.
		Earth, WinterSp., Cas.	Greece, WinterSp., Cas.
		Earth, SnowBoard, Cas.	Greece, SnowBoard, Cas.
		Earth, SnowSki, Cas.	Greece, SnowSki, Cas.
		Olympus, SeaSports, Cas.	

Table 3. The Valid and Invalid compound terms of the example of Figure 1

As the facet *Location* has 8 terms, the facet *Sports* has 7 terms, and the facet *Facilities* has one term, the number of compound terms that contain at most 1 term from each facet is 9*8*2 = 144. This table contains 60 valid and 67 invalid compound terms, thus 127 in total. By adding the (8+7+1=16) singletons (which were omitted from the column of valid) and the empty set we reach the 144.